

What is claimed is:

1 1. In a computer-implemented animation system, a method for animating an object, the method  
2 comprising:

3 receiving a first input, the first input specifying a first behavior, the first behavior

4 indicating how to change a value of a first parameter of the object over time;

5 animating the object by changing the value of the first parameter of the object over time

6 according to the specified behavior; and

7 outputting the animated object.

1 2. The method of claim 1, wherein the object comprises a two-dimensional object.

1 3. The method of claim 1, further comprising receiving a second input, the second input

2 specifying a keyframe indicating the value for the first parameter of the object at a first point in

3 time, and wherein animating the object comprises changing the value of the first parameter of the

4 object according to the specified behavior and further according to the specified keyframe.

1 4. The method of claim 1, further comprising receiving a second input, the second input

2 specifying a second behavior, the second behavior indicating how to change a value of a second

3 parameter of the object over time, and wherein animating the object further comprises changing

4 the value of the second parameter of the object according to the second specified behavior.

1 5. The method of claim 1, further comprising receiving a second input, the second input

2 specifying a second behavior, the second behavior indicating how to change the value of the first

3 parameter of the object over time, and wherein animating the object comprises changing the

4 value of the first parameter of the object according to the first specified behavior and the second

5 specified behavior.

1 6. The method of claim 5, wherein changing the value of the first parameter of the object  
2 according to the first specified behavior and the second specified behavior comprises  
3 determining a combined change to the value of the first parameter of the object according to a  
4 combination of the first specified behavior and the second specified behavior.

1 7. The method of claim 1, wherein the first behavior comprises one from a group consisting of:

2 a Fade In/Fade Out behavior;

3 a Grow/Shrink behavior;

4 a Motion Path behavior;

5 a Snap Alignment to Motion behavior;

6 a Spin behavior;

7 a Throw behavior;

8 an Align to Motion behavior;

9 an Attracted To behavior;

10 an Attractor behavior;

11 a Drag behavior;

12 a Drift Attracted To behavior;

13 a Drift Attractor behavior;

14 an Edge Collision behavior;

15 a Gravity behavior;

16 an Orbit Around behavior;

17 a Random Motion behavior;

18 a Repel behavior;

19 a Repel From behavior;

- 20 a Rotational Drag behavior;
- 21 a Spring behavior;
- 22 a Vortex behavior; and
- 23 a Wind behavior.

1 8. The method of claim 1, wherein the object comprises a text object and the first behavior  
2 comprises one from a group consisting of:

- 3 a Crawl Left behavior;
- 4 a Crawl Right behavior;
- 5 a Scroll Up behavior;
- 6 a Scroll Down behavior;
- 7 a Randomize behavior;
- 8 a Sequence behavior;
- 9 a Position behavior;
- 10 a Rotation behavior;
- 11 an Opacity behavior;
- 12 a Scale behavior;
- 13 a Tracking behavior; and
- 14 a Type On behavior.

1 9. The method of claim 1, wherein the first behavior indicates that the value of the first  
2 parameter of the object should be averaged over time.

1 10. The method of claim 1, wherein the first behavior indicates that the value of the first  
2 parameter of the object should be changed using a user-specified custom change.

- 1 11. The method of claim 1, wherein the first behavior indicates that the value of the first  
2 parameter of the object should be negated.
- 1 12. The method of claim 1, wherein the first behavior indicates that the value of the first  
2 parameter of the object should oscillate over time.
- 1 13. The method of claim 1, wherein the first behavior indicates that the value of the first  
2 parameter of the object should ramp over time.
- 1 14. The method of claim 1, wherein the first behavior indicates that the value of the first  
2 parameter of the object should be randomized.
- 1 15. The method of claim 1, wherein the first behavior indicates that the value of the first  
2 parameter of the object should change over time according to a specified rate.
- 1 16. The method of claim 1, wherein the first behavior indicates that changes to the value of the  
2 first parameter of the object should be executed in reverse order.
- 1 17. The method of claim 1, wherein the first behavior indicates that the value of the first  
2 parameter of the object should not change.
- 1 18. The method of claim 1, wherein the first behavior indicates that the value of the first  
2 parameter of the object should wriggle over time.
- 1 19. The method of claim 1, wherein the object comprises one from a group consisting of:  
2 an image object;  
3 a text object;  
4 a particle system;  
5 a filter;  
6 a generator; and  
7 a behavior.

1 20. The method of claim 1, wherein the first behavior comprises at least one user-settable  
2 behavior parameter, the method further comprising receiving a second input specifying a value  
3 for the behavior parameter, and wherein animating the object comprises changing the value of  
4 the first parameter of the object according to the first specified behavior and the specified value  
5 for the behavior parameter.

1 21. In a computer-implemented animation system, a method for animating an object, the method  
2 comprising:

3 receiving an input, the input specifying the object;  
4 creating one or more duplicates of the object according to a first plurality of parameters;  
5 and  
6 animating the one or more duplicates by changing a value of a parameter of a duplicate  
7 over time according to a second plurality of parameters.

1 22. The method of claim 21, further comprising receiving an input, the input specifying a  
2 parameter, and wherein animating the one or more duplicates comprises changing the value of  
3 the parameter of the duplicate over time according to the second plurality of parameters and the  
4 specified parameter.

1 23. A user interface for a computer program for animating an object according to a behavior, the  
2 behavior having at least one user-settable parameter specifying how the behavior changes a value  
3 of at least one parameter of the object, the user interface comprising:

4 a control area; and  
5 a user-manipulable control element located within the control area, for specifying a value  
6 for the at least one user-settable parameter of the behavior.

1 24. The user interface of claim 23, wherein the user-manipulable control element comprises a  
2 representation of a vector having a magnitude and an orientation.

1 25. The user interface of claim 24, wherein the control area comprises a circle, and wherein the  
2 representation of the vector comprises an arrow, the body of the arrow being a straight line, the  
3 tail of the arrow located in the center of the circle, the arrow pointing toward a point on the  
4 circumference of the circle.

1 26. The user interface of claim 25, wherein the magnitude of the vector controls a first user-  
2 settable parameter of the behavior, and wherein the orientation of the vector controls a second  
3 user-settable parameter of the behavior.

1 27. The user interface of claim 26, wherein the first user-settable parameter comprises a speed  
2 with which the object moves and wherein the second user-settable parameter comprises a  
3 direction in which the object moves.

1 28. The user interface of claim 27, wherein the behavior comprises a Throw behavior.

1 29. The user interface of claim 27, wherein the behavior comprises a Wind behavior.

1 30. The user interface of claim 23, wherein the user-manipulable control element comprises an  
2 arrow having a curved body.

1 31. The user interface of claim 30, wherein the control area comprises a circle, and wherein the  
2 curved body of the arrow comprises an arc of the circumference of the circle, the tail and the  
3 head of the arrow located on the circumference of the circle, the arrow pointing along the  
4 circumference of the circle.

1 32. The user interface of claim 31, wherein the length of the arrow controls a first user-settable  
2 parameter of the behavior, and wherein the direction of the arrow controls a second user-settable  
3 parameter of the behavior.

1 33. The user interface of claim 32, wherein the first user-settable parameter comprises a speed  
2 with which the object rotates and wherein the second user-settable parameter comprises a  
3 direction in which the object rotates.

1 34. The user interface of claim 33, wherein the behavior comprises a Spin behavior.

1 35. The user interface of claim 23, wherein the user-manipulable control element comprises a  
2 first rectangle.

1 36. The user interface of claim 35, wherein the control area comprises a second rectangle, the  
2 second rectangle indicating an original size of the object.

1 37. The user interface of claim 36, wherein a difference between a width of the first rectangle  
2 and a width of the second rectangle controls a first user-settable parameter of the behavior, and  
3 wherein a difference between a height of the first rectangle and a height of the second rectangle  
4 controls a second user-settable parameter of the behavior.

1 38. The user interface of claim 37, wherein the first user-settable parameter comprises a change  
2 in the object's width and the second user-settable parameter comprises a change in the object's  
3 height.

1 39. The user interface of claim 38, wherein the behavior comprises a Grow/Shrink behavior.

1 40. The user interface of claim 23, wherein the user-manipulable control element comprises a  
2 first triangular region and a second triangular region.

1 41. The user interface of claim 40, wherein the control area comprises an area separating the first  
2 triangular region and the second triangular region.

1 42. The user interface of claim 23, wherein a width of the first triangular region controls a first  
2 user-settable parameter of the behavior, and wherein a width of the second triangular region  
3 controls a second user-settable parameter of the behavior.

1 43. The user interface of claim 42, wherein the first user-settable parameter comprises a fade-in  
2 time of the object and the second user-settable parameter comprises a fade-out time of the object.

1 44. The user interface of claim 43, wherein the behavior comprises a Fade In/Fade Out behavior.

1 45. The user interface of claim 23, wherein the control area is semi-transparent.

1 46. A user interface for a computer program for animating an object, wherein animating an  
2 object comprises creating one or more duplicates of the object and animating the one or more  
3 duplicates by changing a value of a parameter of the one or more duplicates over time, the user  
4 interface comprising:

5 a control area; and

6 one or more controls for setting one or more parameters of the animation.

1 47. The user interface of claim 46, wherein the user-manipulable control element comprises a  
2 representation of a vector having a magnitude and an orientation.

1 48. The user interface of claim 47, wherein the control area comprises a circle, and wherein the  
2 representation of the vector comprises an arrow, the body of the arrow being a straight line, the  
3 tail of the arrow located in the center of the circle, the arrow pointing toward a point on the  
4 circumference of the circle.

1 49. The user interface of claim 48, wherein the magnitude of the vector controls a first user-  
2 settable parameter of the animation, and wherein the orientation of the vector controls a second  
3 user-settable parameter of the animation.

1 50. The user interface of claim 49, wherein the first user-settable parameter comprises a speed  
2 with which the one or more duplicates moves and wherein the second user-settable parameter  
3 comprises a direction in which the one or more duplicates moves.



1 51. The user interface of claim 46, wherein the user-manipulable control element comprises two  
2 points.

1 52. The user interface of claim 47, wherein the control area comprises a circle, and wherein the  
2 two points are located on the circumference of the circle, and wherein the two points specify a  
3 segment of the circle.

1 53. The user interface of claim 48, wherein the size of the segment of the circle controls a first  
2 user-settable parameter of the animation, and wherein the position of the segment of the circle  
3 controls a second user-settable parameter of the animation.

1 54. The user interface of claim 49, wherein the first user-settable parameter comprises a size of a  
2 range in which the one or more duplicates moves and wherein the second user-settable parameter  
3 comprises a location of the range in which the one or more duplicates moves.

1 55. The user interface of claim 46, wherein the control area is semi-transparent.

1 56. A method for generating a frame of an object using behaviors, comprising:

2 determining a current state of the object;  
3 traversing a data structure to identify behaviors affecting the object;  
4 accumulating forces for the behaviors affecting the object; and  
5 generating a frame of the object according to the accumulated forces.

1 57. The method of claim 56, further comprising determining an initial velocity for the object.

1 58. The method of claim 56, wherein at least one of the behaviors is a motion behavior.

1 59. The method of claim 56, wherein at least one of the behaviors is a simulation behavior.

1 60. The method of claim 56, wherein at least one of the behaviors is a parameter behavior.

1 61. The method of claim 56, wherein the data structure comprises a tree structure.

- 1 62. The method of claim 56, wherein generating the frame comprises applying a mid-point  
2 method differential solver to determine a new parameter value for the object.
- 1 63. The method of claim 56, wherein the parameter value comprises a position of the object.
- 1 64. The method of claim 56, further comprising:  
2       traversing the data structure to identify collisions; and  
3       responsive to the existence of a collision, adjusting a system state to maintain a collision  
4       constraint.
- 1 65. The method of claim 56, further comprising iteratively repeating the animating step until a  
2 desired frame is reached.
- 1 66. The method of claim 56, wherein at least one object state is specified in terms of a keyframe,  
2 the method further comprising converting at least one keyframe into a set of forces that, when  
3 applied to the object, approximate the motion represented by the keyframe.
- 1 67. The method of claim 66, wherein converting at least one keyframe into a set of forces  
2 comprises deriving a set of forces based on the velocity and acceleration at the keyframe.
- 1 68. A method for generating an animation for an object using behaviors, the animation  
2 comprising a plurality of frames, the method comprising:  
3       for each frame:  
4           determining a current state of the object;  
5           traversing a data structure to identify behaviors affecting the object;  
6           accumulating forces for the behaviors affecting the object;  
7           generating a frame of the object according to the accumulated forces; and  
8           outputting the generated frame.

1 69. The method of claim 68, wherein at least one of the determining, traversing, accumulating,  
2 generating and outputting steps for a first frame is performed concurrently with at least one of  
3 the determining, traversing, accumulating, generating and outputting steps for a second frame.

1 70. A method for animating an object using parameter behaviors, comprising:

2 traversing a stack of operations on a range of values;

3 responsive to a single behavior value being sufficient to evaluate all operations in the

4 stack, passing the single behavior value to each operation in the stack; and

5 responsive to a single behavior value not being sufficient to evaluate all operations in the

6 stack:

7 determining a range of input values to compute a requested output range; and

8 passing the determined range of input values to each operation in the stack.

1 71. A method for animating an object using a behavior, comprising:

2 outputting an original animation for the object according to a first behavior;

3 concurrently with outputting the object animation, accepting user input; and

4 outputting an updated animation for the object according to the user input.

1 72. The method of claim 71, wherein the user input comprises a command for changing a value  
2 of a parameter of the behavior, and wherein outputting the updated animation comprises  
3 outputting the updated animation according to the changed value of the parameter.

1 73. The method of claim 71, wherein the user input comprises a command for applying a second  
2 behavior to the object and wherein outputting the updated animation comprises outputting the  
3 updated animation according to the first and second behaviors.

1 74. The method of claim 71, wherein outputting the updated animation is performed without  
2 interrupting the animation for the object.

1 75. The method of claim 71, wherein the updated animation reflects the changed value of the  
2 parameter in real-time.

1 76. The method of claim 71, wherein outputting the original animation and outputting the  
2 updated animation each comprise rendering a plurality of frames and caching the rendered  
3 frames.

1 77. The method of claim 71, wherein outputting the original animation and outputting the  
2 updated animation each comprise rendering each of a plurality of frames sequentially.

1 78. The method of claim 71, wherein outputting the original animation and outputting the  
2 updated animation each comprise rendering each of a plurality of frames sequentially by  
3 calculating a current frame based on a previous frame.

1 79. The method of claim 71, wherein outputting the original animation and outputting the  
2 updated animation each comprise rendering a plurality of frames and periodically caching a  
3 subset of the rendered frames in an interval cache.

1 80. The method of claim 71, wherein outputting the original animation and outputting the  
2 updated animation each comprise evaluating, by a first thread, a first subset of frames, and  
3 evaluating, by a second thread, a second subset of frames.

1 81. The method of claim 80, wherein the first subset and the second subset of frames each  
2 comprise alternate frames of the animation.

1 82. In a computer-implemented animation system, a method for animating an object, the method  
2 comprising:

3 receiving a first input, the first input specifying a first behavior, the first behavior

4 indicating how to change a value of a parameter of the object over time;

- 5 using at least one of a pixel shader and a vertex shader to generate a plurality of frames of
- 6 the object, according to the specified behavior; and
- 7 outputting the plurality of frames.